

Software Testing

Teststufen, Testphasen und Zusammenhänge

23.Dezember 2016

Stefan Berger

HWZ

Die Hochschule für Wirtschaft
in Zürich

Motivation

Warum testen wir ?

1. Testen soll Fehler finden
2. Fehlerfreie Software herstellen
3. Hohe Software - Qualität

Hohe Software-Qualität

Wie erreichen wir hohe Software-Qualität ?

- **Statisches Testen** → manuelle Überprüfung
 - Code – Analyse durch formale Prüfung
 - Code – Review → frühe Fehlererkennung
- **Dynamisches Testen** → verlangt Ausführung der Software
 - Unittest, Integrationstest, Systemtest usw.
- **Test nach Disziplinen** → Performancetest, Lastentest, Crashtest, etc.

Was ist ein Test ?

- Prüft Korrektheit eines Softwarebausteins
- Jederzeit und beliebig oft wiederholbar
- Ergebnis bei gleichen Parametern gleich
- Reduziert das Restrisiko verbleibender Fehler
- Deckt Fehler auf, beweist nicht Fehlerfreiheit

Testen versus Debugging

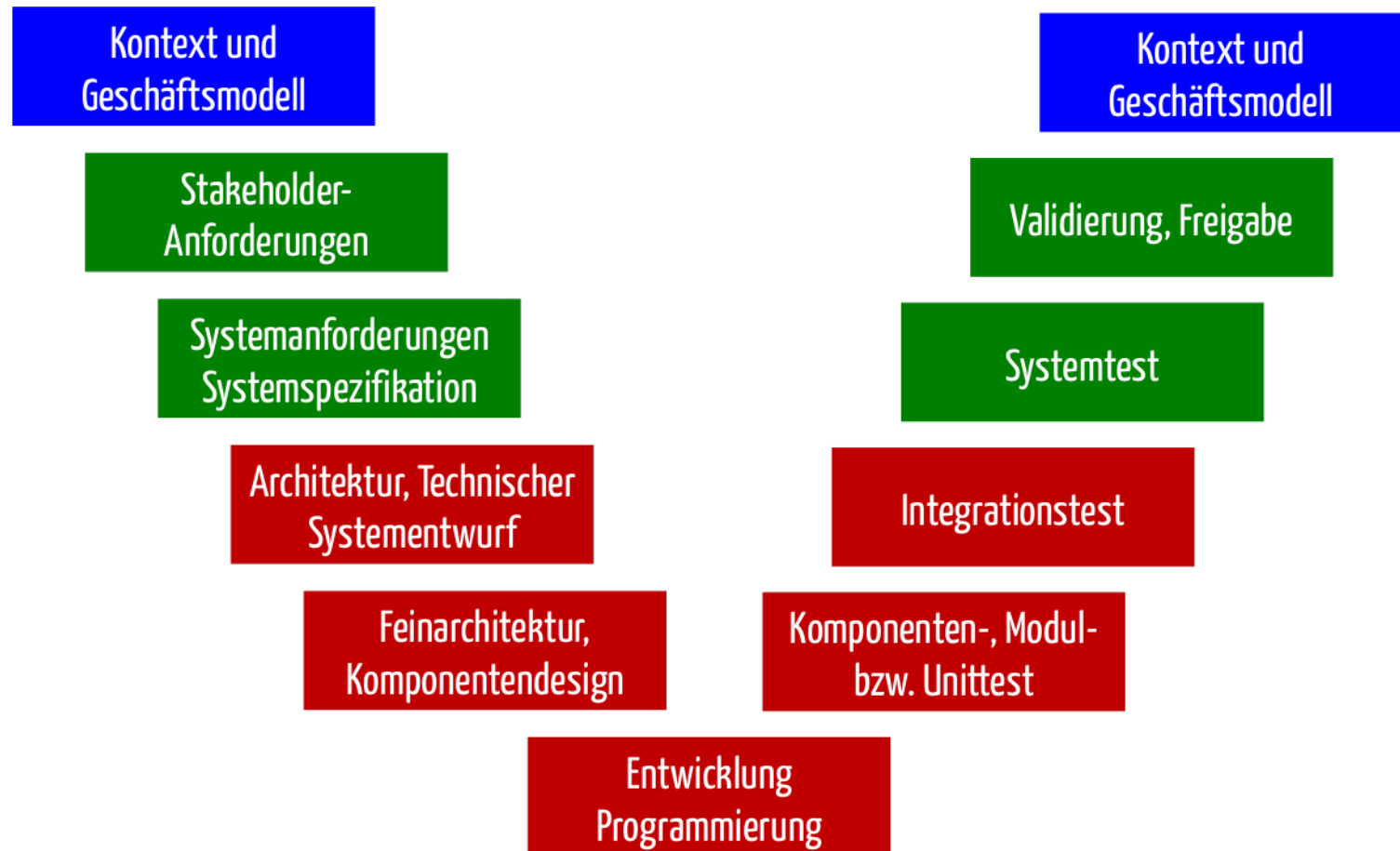
Testen

→ kann Fehlerwirkungen zeigen, die durch Fehler verursacht werden

Debugging

→ ist eine Entwicklertätigkeit, die Ursache eines Fehlers identifiziert

Teststufen – Übersicht / V-Model



Testarten

- UnitTest (kleinste Einheit testen)
- Integrationstest (Testen des Zusammenspiels von Komponenten)
- Systemtest (das ganze System testen)
- Akzeptanztest (Benutzertest)
- Performancetest (Zeitmessungstest“)

Unittest

Modultest oder Komponententest

Was: Einzelne Teile einer Software (Klasse, Funktion, Unterprogramm, Module) wird abgegrenzt und abgeschlossen getestet

Ziel: Entwickler „beweist“ technische und fachliche Korrektheit

Integrationstest

Was: Fertige Komponenten werden im Zusammenspiel getestet

Ziel: Zusammenspiel der Komponenten funktionieren korrekt

Systemtest

Was: In der Systemtest – Umgebung wird unter produktiven Konditionen getestet. Der Test wird durch die Entwickler durchgeführt

Ziel: Der Systemtest zeigt die Eignung für den produktiven Einsatz

Akzeptanztest

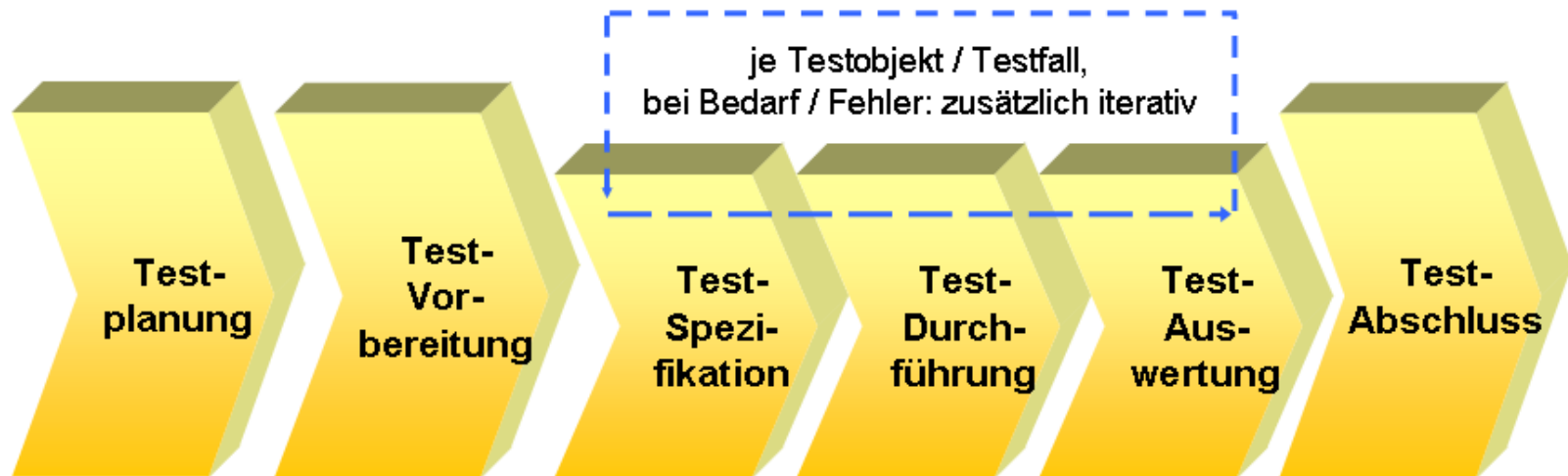
Abnahmetest, Verfahrenstest, oder auch Kunden Akzeptanz Test

Was: Der Kunde testet das Software – Verhalten

Ziel: Der Kunde übernimmt die Software in den Betrieb.

Testphasen - Übersicht

Das Phasenmodell beim Testen



Quelle: Pol, Koomen, Spillner

Das Modell ist 'generisch', d.h.:

Es wird in mehreren Ebenen, mit unterschiedlichen Schwerpunkten angewendet.

a) Projekt-Ebene, Masterplanung:

b) Je Teststufe:

c) Je Testobjekt / Testfall:

Testplanung

Wann: Testplanung von Anfang aufbauen – bei Projektstart

Wer: sämtliche beteiligte Personen (Entwickler, Berater und Kunden)

Was: eingesetzte Werkzeuge, Testdaten, Umgebung, Testumfang, gewünschte Resultate, Testorganisation definieren, Häufigkeit

Testvorbereitung

Ziel:

- Ausgangslage gemäss Plan aufbauen
- wiederverwendbare Ausgangslage automatisieren
- Testdaten und Schnittstellen bereitstellen
- User / User Berechtigungen vorbereiten

Testspezifikation

Ziel:

- Beschreiben jedes Testfalls
- Ausgangslage und Bedingung (Daten, Position, Eingabe, Rollen)
- Aktionsbeschreibung
- Soll – Ergebnis spezifizieren
- Ist – Ergebnis festhalten
- Resultat festhalten

Testfall - Vorlage

	A	B	C	D	E	F	G	H	I	J	K
1	Test Case ID		BU_001	Test Case Description	Test the Login Functionality in Banking						
2	Created By		Mark	Reviewed By	Bill			Version	2.1		
3											
4	QA Tester's Log	Review comments from Bill incorporate in version 2.1									
5											
6	Tester's Name		Mark	Date Tested	1-Jan-2017			Test Case (Pass/Fail/Not	Pass		
7											
8	S #	Prerequisites:				S #	Test Data				
9	1	Access to Chrome Browser				1	Userid = mg12345				
10	2					2	Pass = df12@434c				
11	3					3					
12	4					4					
13											
14	Test Scenario	Verify on entering valid userid and password, the customer can login									
15											
16	Step #	Step Details		Expected Results		Actual Results		Pass / Fail / Not executed / Suspended			
17											
18	1	Navigate to http://demo.guru99.com		Site should open		As Expected		Pass			
19	2	Enter Userid & Password		Credential can be entered		As Expected		Pass			
20	3	Click Submit		Cutomer is logged in		As Expected		Pass			
21	4										

<
>
Test Case 1
Test Case 2
Test Case 3
+
:
<

Testdurchführung

Ziel:

Durchführung aller Testfälle (automatisiert) bzw. einzeln nach Drehbuch
In der Regel ist der Entwickler nicht mehr der Tester

Testauswertung

- Festhalten aller Testfälle und auswerten
- Fehler – Zuordnen nach Klassifizierung
- Testfälle dokumentiert archivieren
- dient als Input als Korrektur

Testabschluss

- Alle Tests abschliessen auf allen Stufen
 - Input für die Zukunft
 - Dokumentation für die Entwicklung / Projektmanagement

Test - Software

- <http://www.seleniumhq.org/> (testen von Web – UI)
- <http://junit.org/> (testen von Java)
- <https://techbeacon.com/6-top-open-source-testing-automation-frameworks-how-choose>
- und viele mehr...